Triplet Graph Convolutional Network for Multi-scale Analysis of Functional Connectivity Using Functional MRI

Dongren Yao[1,2,3], Mingxia Liu[3(✉)], Mingliang Wang[3,4], Chunfeng Lian[3], Jie Wei[3,5], Li Sun[6], Jing Sui[1,2(✉)], and Dinggang Shen[3(✉)]

[1] Brainnetome Center & National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China
jing.sui@nlpr.ia.ac.cn
[2] University of Chinese Academy of Sciences, Beijing 100049, China
[3] Department of Radiology and BRIC, University of North Carolina at Chapel Hill, Chapel Hill, NC 27599, USA
{mxliu,dgshen}@med.unc.edu
[4] College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, China
[5] School of Computer Science, Northwestern Polytechnical University, Xi'an 710072, China
[6] National Clinical Research Center for Mental Disorders & Key Laboratory of Mental Health, Ministry of Health, Peking University, Beijing 100191, China

Abstract. Brain functional connectivity (FC) derived from resting-state functional MRI (rs-fMRI) data has become a powerful approach to measure and map brain activity. Using fMRI data, graph convolutional network (GCN) has recently shown its superiority in learning discriminative representations of brain FC networks. However, existing studies typically utilize one specific template to partition the brain into multiple regions-of-interest (ROIs) for constructing FCs, which may limit the analysis to a single spatial scale (i.e., a fixed graph) determined by the template. Also, previous methods usually ignore the underlying high-order (e.g., triplet) association among subjects. To this end, we propose a multi-scale triplet graph convolutional network (MTGCN) for brain functional connectivity analysis with rs-fMRI data. Specifically, we first employ multi-scale templates for coarse-to-fine ROI parcellation to construct multi-scale FCs for each subject. We then develop a triplet GCN (TGCN) model to learn multi-scale graph representations of brain FC networks, followed by a weighted fusion scheme for classification. Experimental results on 1,218 subjects suggest the efficacy or our method.

1 Introduction

Resting-state functional magnetic resonance imaging (fMRI) provides a powerful tool for capturing brain functional connectivity even between spatially-remote

D. Zhang et al. (Eds.): GLMI 2019, LNCS 11849, pp. 70–78, 2019.
https://doi.org/10.1007/978-3-030-35817-4_9

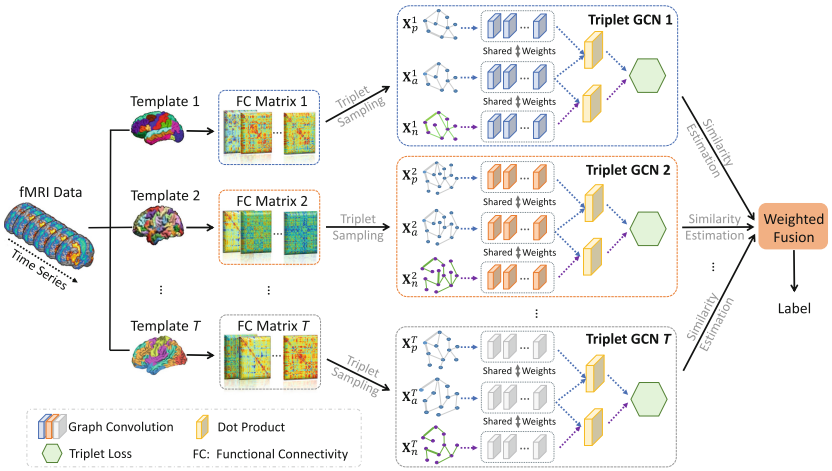

Fig. 1. Illustration of our Multi-scale Triplet Graph Convolutional Network (MTGCN). Multiple templates are used for coarse-to-fine parcellation of brain regions and construction of functional connectivity matrices/graphs. Each triplet GCN (TGCN) inputs a triplet of three graphs (e.g., $\mathbf{X}_a^1$, $\mathbf{X}_p^1$, and $\mathbf{X}_n^1$) with the same architecture but different signals, and outputs similarity among the triplet. A weighted fusion scheme is used to combine estimated labels generated from multi-scale TGCNs for classification.

brain regions. Recent studies have shown that fMRI-based analysis of brain connectivity is effective in helping understand the pathology of brain diseases, such as autism spectrum disorder (ASD) and attention-deficit/hyperactivity disorder (ADHD) [1,2]. Since functional disorder (e.g., abnormality or dysfunction between multiple regions) of brain networks has been found in these diseases [3], many studies have focused on using fMRI-based brain connectivity networks to explore potential disorder patterns of brain diseases characterized between patients and healthy controls (HCs). In fMRI-based studies, each brain can be represented as an undirected or directed graph/network containing functionally interconnected regions-of-interests (ROIs), with each ROI treated as a vertex.

Convolutional neural networks (CNNs) have shown their superiority in analyzing functional connectivity (FC) networks [4], but ignoring the important topology information. Graph convolutional neural networks (GCNs) integrate graph convolutional layers to explicitly capture topological information, which can learn useful representations of brain FC networks for brain disease classification [5]. However, existing GCN-based methods usually use one specific template for brain ROI parcellation, which limits the analysis to a single spatial scale (i.e., a fixed graph) determined by the template. Also, previous studies simply model the pairwise relationship between graphs/subjects, without considering the triplet similarity among subjects.

To address these issues, we propose a multi-scale triplet graph convolutional network (MTGCN) for brain functional connectivity analysis with rs-fMRI data,

as illustrated in Fig. 1. Specifically, we first apply multi-scale templates to each subject for generating multi-scale FC networks under different spatial scales and ROI definitions. We then develop a triplet GCN (TGCN) model to automatically extract multi-scale graph representations of brain FC networks, incorporated by the triplet similarity among subjects (e.g., $\mathbf{X}_a^1$, $\mathbf{X}_p^1$, and $\mathbf{X}_n^1$). A weighted fusion scheme is finally used to combine estimated labels generated from multi-scale TGCNs for brain disease classification. Experiments results on 1,218 subjects from two real datasets demonstrate the effectiveness of our MTGCN method in ASD and Adult ADHD identification, compared with state-of-the-art methods.

2 Method

Multi-scale FC Network: To overcome the spatial limitation of using only a single template for ROI parcellation and FC construction, we employ T ($T \geq 2$) templates (with coarse-to-fine ROI definitions) to generate multi-scale FC networks for each subject. Given a specific template, one can construct a graph (i.e., FC matrix) for each subject, where each node represents an ROI, and the connectivity between a pair of ROIs is represented by the Pearson correlation of their mean time series signals. That is, a *fully-connected* graph can be constructed for each subject, with each vertex connected with all the other vertices.

In this work, we would like to employ spectral GCN for analyzing brain function connectivity. Graph convolution is a type of Laplacian smoothing that computes the new features of a vertex as the weighted average of itself and its neighbors [6]. Using fully-connected network as graph topology, the smoothing operation will make features of those 'connected' vertices similar, thus failing to capture the *node-centralized local topology* via spectral GCNs. Therefore, we propose to construct a KNN graph (other than a fully-connected graph), by connecting each vertex with its k-nearest neighbors to model the node-centralized local topology. Also, we aim to capture shared graph topology among all studied subjects, and thus need to construct a KNN graph at the *group-level* (rather than subject-level). Specifically, we first calculate the mean FC matrix of all training subjects in a template space, and then construct a KNN graph by connecting each vertex with its k-nearest neighbors (measured by the Pearson correlation). Hence, the graph topology (reflected by vertices and their connectivity) of such a group-level KNN graph is shared by all subjects. For a subject, the signal of each vertex in a KNN graph is represented by an n-dimensional vector (i.e., the Pearson correlations between this vertex and all the other ones) corresponding to the n-th row of FC matrix, and n is the number of vertices/ROIs defined by the template. Hence, we can represent each subject by both its graph topology and vertex features/signals in specific template space. Given T templates, we can generate T KNN graphs, with each graph denoting the group-level topology of FC networks at a specific spatial scale.

Triplet GCN: Graph convolutional network (GCN) generalizes traditional convolutional neural network from Euclidean data (e.g., 2D or 3D images) to the

non-Euclidean domain (e.g., graphs and manifolds), and has been emerging as a promising method for graph mining [7]. Denote a graph as $\mathbf{X} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is the set of vertices and $\mathcal{E}$ is the set of edges. Also, an adjacency matrix $\mathbf{A} = [a_{ij}] \in \mathbb{R}^{n \times n}$ encodes the connectivity among vertices, with the element a_{ij} indicating whether the i-th and j-th vertices are connected ($a_{ij} = 1$) or not ($a_{ij} = 0$). Denote $\mathbf{D} = diag(d_1, d_2, \cdots, d_n)$ as a degree matrix, with each element $d_i = \sum_j a_{ij}$ denoting the number of edges connected to the i-th vertex. Spectral GCN defines the convolution by decomposing a graph signal $s \in \mathbb{R}^n$ (defined on the vertex of graph $\mathbf{X}$) in the spectral domain. Then the signal s will be processed by a spectral filter δ_θ with the first order polynomial of ChebyNet [8], instead of explicitly computing the Laplacian eigenvectors. To reduce the number of parameters (i.e., θ_0 and θ_1), the spectral GCN model assumes that $\theta = \theta_0 = -\theta_1$, and the graph convolution is correspondingly defined as:

$$\delta_\theta * s \approx \theta_0 s - \theta_1 \mathbf{D}^{-\frac{1}{2}} \mathbf{A} \mathbf{D}^{-\frac{1}{2}} s = \theta(\mathbf{I}_n + \mathbf{D}^{-\frac{1}{2}} \mathbf{A} \mathbf{D}^{-\frac{1}{2}}) s \tag{1}$$

where $\mathbf{I} \in \mathbb{R}^{n \times n}$ is an identity matrix. With each convolution defined in Eq. 1 followed by a non-linear activation function (i.e., ReLU), one can generate a simple and flexible spectral GCN by stacking multiple convolutional layers.

Recently, a method based on spectral GCN has been applied to analyze brain FC network for ASD classification, in which two GCN models were constructed to measure the similarity between paired graphs [5]. However, this method only models the pairwise relationship of graphs, ignoring the high-order (e.g., triplet) association among graphs/subjects. To mine the potential high-order similarity between *inter-class* and *intra-class* subjects, we develop a triplet GCN (TGCN) module containing 3 parallel subnetworks (with 3 graph convolutional layers), and these subnetworks share the same network parameters. As shown in Fig. 1, the input of the t-th TGCN module is a triplet $\{\mathbf{X}_a^t, \mathbf{X}_p^t, \mathbf{X}_n^t\}$ containing an *anchor* subject $\mathbf{X}_a^t$, a *positive* sample $\mathbf{X}_p$, and a *negative* sample $\mathbf{X}_n^t$, where $\mathbf{X}_a^t$ and $\mathbf{X}_p^t$ are samples from the same category and $\mathbf{X}_n^t$ is selected from a different category. Note that samples in each triplet share the same graph architecture but represented by different vertex signals. Using TGCN, we encourage that the learned graph representations of the anchor and positive samples belonging to the same class (e.g., ASD patient) are similar, while representations of the anchor and negative samples belonging to a different category (e.g., HC) are dissimilar. The loss function to train the t-th TGCN module is defined as

$$L_t = \sum_{m=1}^{M} \left[\| g_t(\mathbf{X}_{a,m}^t) - g_t(\mathbf{X}_{p,m}^t) \|_2^2 - \| g_t(\mathbf{X}_{a,m}^t) - g_t(\mathbf{X}_{n,m}^t) \|_2^2 + \lambda_{trp} \right]_+ \tag{2}$$

where M is the number of input triplets, $[\cdot]_+ = max(s, 0)$, $g_t(\cdot)$ denotes the embedding generated by the t-th TGCN module, and the margin λ_{trp} is used to enforce the distance between the anchor and negative samples. The output of TGCN is the similarity of the anchor $\mathbf{X}_a$ belonging to a specific category. Given T templates, a multi-scale network containing T TGCN modules can be obtained (see Fig. 1), with each one corresponding to a specific template.

Weighted Fusion: With a single-scale TGCN based on a specific template, we can measure the triplet similarity between an anchor sample and two samples from two different categories. In this work, we hypothesize that *similar subjects tend to remain similar in multi-scale template spaces*. With this assumption, the outputs of T TGCN modules are fused via a weighted fusion strategy as follows:

$$L_{Fusion} = \sum_{t=1}^{T} \gamma_t L_t \tag{3}$$

where $\gamma_t = \omega_t / \sum_{t=1}^{T} \omega_t$ represents the weight of the t-th TGCN module, and ω_t is the classification accuracy achieved by the t-th TGCN on the training data.

Implementation: In the *training* stage, for each training subject (anchor) $\mathbf{X}_a$, we randomly select a pair of positive and negative subjects from training set to generate a triplet set $\{\mathbf{X}_{a,q}, \mathbf{X}_{p,q}, \mathbf{X}_{n,q}\}_{q=1}^{Q}$. Due to limited training subjects, such random selection process is repeated $Q = 25$ to augment samples. In the *testing* stage, for a testing subject (anchor), we first randomly select 5 pairs of positive and negative subjects from training set [9], yielding 5 triplets as the input of MTGCN. The prediction results of these 5 triplets are ensembled to get a final label for the testing subject. Such process is used to suppress the bias introduced by random selection of positive and negative training subjects.

3 Experiment and Results

Subjects and Image Pre-processing: Two datasets with resting-state fMRI data are used for performance evaluation in this work. The first one is an Autism dataset with 485 ASD patients and 544 healthy controls (HCs) from 34 imaging sites in ABIDE-I and ABIDE-II, with considerable data distribution diversity. The largest site contains 101 subjects, while the smallest one only has 11 subjects. Another is an Attention Hyperactivity Deficit Disorder (ADHD) dataset, containing 112 ADHD patients and 77 age-matched HCs recruited from the Sixth Hospital of Peking University (PKU6).

Each fMRI scan was pre-processed using the Data Processing Assistant for Resting-State fMRI (DPARSFA). The processing pipeline is as follows: (1) discarding the first ten volumes, (2) slice timing correction, (3) head motion correction, (4) normalization with an EPI template in the MNI space, resampling to $3 \times 3 \times 3\,\text{mm}^3$ resolution, (5) spatial smoothing using a 4 mm full width half maximum Gaussian kernel, (6) linear detrend and temporal band-pass filtering (0.01 Hz–0.1 Hz), (7) regression out nuisance signals of head motion parameters, white matter, cerebrospinal fluid (CSF), and global signals. The registered fMRI volumes were partitioned into ROIs according to different templates. For the Autism dataset, we use preprocessed data, i.e., FC networks constructed on $T = 3$ predefined templates, including Bootstrap Analysis of Stable Cluster parcellation with 122 ROIs (BASC122) and 197 ROIs (BASC197), and Power Template with 264 ROIs (Power264). For the ADHD dataset, we preprocessed the fMRI time series data using $T = 2$ templates, i.e., Automated Anatomical

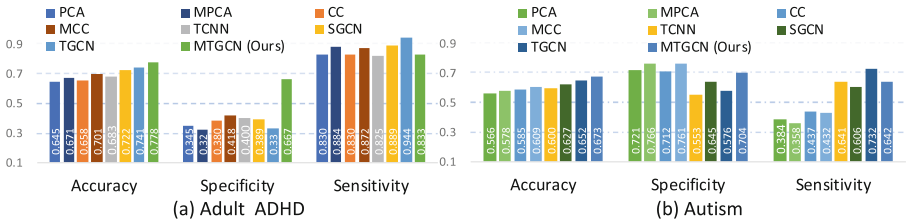

Fig. 2. Results achieved by different methods in (a) Adult ADHD vs. HC classification and (b) ASD vs. HC classification.

Labelling (AAL) with 116 ROIs and Brainnetome template (Brainnetome) with 273 ROIs, aiming to partition the brain in a coarse-to-fine manner. In each template space, we extracted the mean time series within each ROI to construct the original FC network/matrix for each subject, and the connectivity was represented by the Pearson correlation coefficient between pairs of ROI-based time series (normalized to z score using Fisher transformation). Then, the constructed FC matrix was used for the construction of multi-scale KNN graphs in MTGCN.

Experimental Setting: Two classification tasks are performed in the experiments, i.e., (1) ASD vs. HC classification and (2) Adult ADHD vs. HC classification. We use 60% of the data for training, 10% for validation, and the remaining 30% for testing, and such process is repeated 5 times. Three evaluation metrics are used, including classification accuracy, sensitivity, and specificity. We first compare MTGCN with 2 baseline methods and their multi-template variants, i.e., (1) single-template Principal Component Analysis (**PCA**) with a random forest classifier; (2) single-template clustering coefficient (**CC**) with RF as the classifier; (3) multi-template PCA (**MPCA**); and (4) multi-template CC (**MCC**). In PCA, PCA is used for dimension reduction based on the original FC matrix of each subject in a template space, where the number of components is chosen from 3 to 50, with the step size of 1 via two-fold inner CV. In CC, CC features measure the clustering degree of each node in a graph/matrix in a template space. The number of trees in RF is chosen from $[50, 200]$ (step size: 10), and the depth of each tree is selected from $[1, 6]$ (step size: 1).

We further compare MTGCN with two state-of-the-art methods, i.e., (1) CNN with our proposed triplet representation learning strategy (**TCNN**), and (2) Siamese GCN (**SGCN**) [5]. To validate our multi-scale learning strategy, we further compare MTGCN with its single-scale variant (**TGCN**) using one template. Similar to MTGCN, TCNN inputs the original FC matrix of each subject and aims to learn graph embeddings by measuring the similarity among three subjects, and the network contains 3 convolutional layers and two fully connected layers. These 3 convolutional layers in TCNN have 4, 8, and 8 channels, and the corresponding kernels have the sizes of 5×5, 4×4, and 3×3, respectively. The SGCN method [5] can learn graph representations of brain FC networks. For a fair comparison, both SGCN and TGCN methods share the same GCN architec-

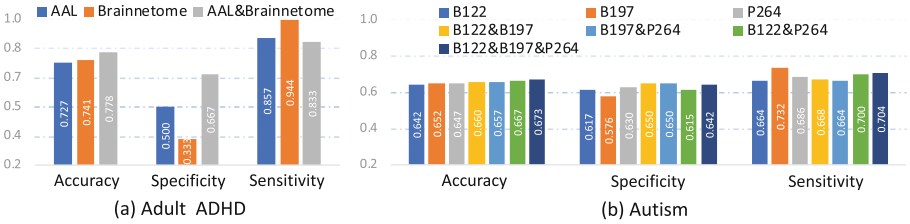

Fig. 3. Influence of templates on the performance of our MTGCN method in (a) Adult ADHD vs. HC classification and (b) ASD vs. HC classification. The symbol & represents template combination. B122: BASC122; B197: BASC197; P264: Power264.

ture as MTGCN. Such process is used to suppress the bias introduced by random selection of positive and negative subjects. In MTGCN and TGCN, the k value is chosen from $[5, 30]$ with the step size of 5 for KNN graph construction, and the parameter λ_{trp} in Eq. (2) is set as 5 empirically. Four deep methods (i.e., TCNN, SGCN, TGCN and MTGCN) are optimized using the ADAM algorithm (learning rate: 0.001; epoch: 100; mini-batch size: 50). Since five competing methods (i.e., PCA, CC, TCNN, SGCN, and TGCN) can use only one template, we perform template selection on the validation data. For MPCA and MCC, the same multiple templates are used as those in our MTGCN method.

Results: The disease classification results are reported in Fig. 2. From Fig. 2, one can find several interesting observations. *First*, four deep learning methods, i.e., TCNN, SGCN, TGCN and MTGCN that learn task-oriented features of FC networks, consistently achieve better performance than four methods (i.e., PCA, CC, MPCA and MCC) using handcrafted features. *Second*, GCN methods (i.e., SGCN, TGCN, and MTGCN) that explicitly model graph topology generally outperform the TCNN method, suggesting the usefulness of mining graph topology of FC networks to boost the classification performance. *Besides*, compared with SGCN based on the pairwise relationship among subjects, our MTGCN method that models the triplet relationship achieves at least 5% improvement (w.r.t., accuracy) in both datasets. It implies that modeling the high-order (e.g., triplet) relationship among subjects boosts the classification results. *Furthermore*, MTGCN usually yields better performance than its single-template variant (i.e., TGCN), while MPCA and MCC using multiple templates outperforms their single-template versions (i.e., PCA and CC). This implies that the efficacy of our multi-scale learning strategy using multiple templates for ROI partition.

Discussion: We further analyze the effect of different templates and template combinations used in MTGCN, with results shown in Fig. 3. The Adult ADHD dataset has two templates and 3 combinations, while the Autism dataset has 3 templates and 7 template combinations. Figure 3 suggests that using multi-scale templates in MTGCN usually yields better performance, compared with

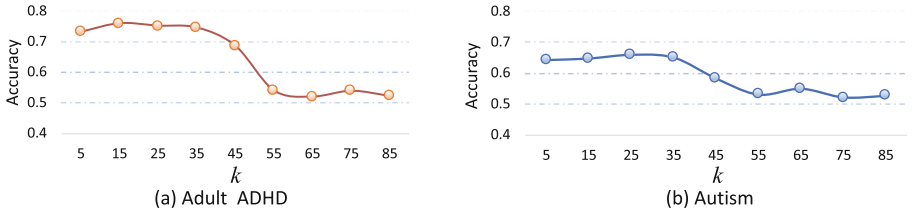

Fig. 4. Influence of the parameter k for constructing KNN graphs on the performance of MTGCN in (a) Adult ADHD vs. HC classification and (b) ASD vs. HC classification.

that using a single template. We further analyze the influence of k in KNN graph construction in MTGCN, with results given in Fig. 4. Figure 4 suggests that MTGCN can achieve good results in both datasets when the value of k is within the range $[15, 35]$, and using larger k values (e.g., >45) in MTGCN will degrade the performance. The possible reason is that MTGCN cannot focus on the node-centralized local topology when using a too large value of k. As the future work, we plan to use unified templates on these 2 datasets for further analysis.

4 Conclusion

We proposed a multi-scale triplet graph convolutional network (MTGCN) for fMRI-based brain disease diagnosis. We first constructed multi-scale functional connectivity networks for each subject based on multiple templates (with different ROI partitions). We then designed a multi-scale triplet GCN (TGCN) model to learn graph representation for each subject, followed by a weighted fusion strategy to fuse the outputs of multiple TGCNs for classification. Experimental results on two real fMRI datasets suggest the efficacy of MTGCN.

References

1. Zhang, D., Huang, J., Jie, B., Du, J., Tu, L., Liu, M.: Ordinal pattern: a new descriptor for brain connectivity networks. IEEE Trans. Med. Imaging **37**(7), 1711–1722 (2018)
2. Lian, C., et al.: Multi-channel multi-scale fully convolutional network for 3D perivascular spaces segmentation in 7T MR images. Med. Image Anal. **46**, 106–117 (2018)
3. Norman, L.J., et al.: Structural and functional brain abnormalities in attention-deficit/hyperactivity disorder and obsessive-compulsive disorder: a comparative meta-analysis. JAMA Psychiatry **73**(8), 815–825 (2016)
4. Zhao, Y., et al.: Automatic recognition of fMRI-derived functional networks using 3-D convolutional neural networks. IEEE Trans. Biomed. Eng. **65**(9), 1975–1984 (2018)
5. Ktena, S.I., et al.: Metric learning with spectral graph convolutions on brain connectivity networks. NeuroImage **169**, 431–442 (2018)

6. Li, Q., Han, Z., Wu, X.M.: Deeper insights into graph convolutional networks for semi-supervised learning. In: AAAI (2018)
7. Wu, Z., Pan, S., Chen, F., Long, G., Zhang, C., Yu, P.S.: A comprehensive survey on graph neural networks. arXiv preprint: arXiv:1901.00596 (2019)
8. Kipf, T.N., Welling, M.: Semi-supervised classification with graph convolutional networks. arXiv preprint: arXiv:1609.02907 (2016)
9. Wu, C.Y., Manmatha, R., Smola, A.J., Krahenbuhl, P.: Sampling matters in deep embedding learning. In: ICCV, pp. 2840–2848 (2017)